

# Introduction to Machine Learning

## Gaussian Processes Basics



$$\begin{matrix} f(x) \\ \text{■} \\ \text{■} \\ \vdots \\ \text{■} \\ \text{■} \\ \text{■} \end{matrix} \sim \mathcal{N}(\mu, \Sigma)$$

### Learning goals

- GPs model distributions over functions
- The marginalization property makes this distribution easily tractable
- GPs are fully specified by mean and covariance function
- GPs are indexed families

# WEIGHT-SPACE VIEW

- Until now we considered a hypothesis space  $\mathcal{H}$  of parameterized functions  $f(\mathbf{x} \mid \theta)$  (in particular, the space of linear functions).
- Using Bayesian inference, we derived distributions for  $\theta$  after having observed data  $\mathcal{D}$ .
- Prior beliefs about the parameter are expressed via a prior distribution  $q(\theta)$ , which is updated according to Bayes' rule

$$\underbrace{p(\theta|\mathbf{X}, \mathbf{y})}_{\text{posterior}} = \frac{\overbrace{p(\mathbf{y}|\mathbf{X}, \theta)}^{\text{likelihood}} \overbrace{q(\theta)}^{\text{prior}}}{\underbrace{p(\mathbf{y}|\mathbf{X})}_{\text{marginal}}}.$$



# FUNCTION-SPACE VIEW

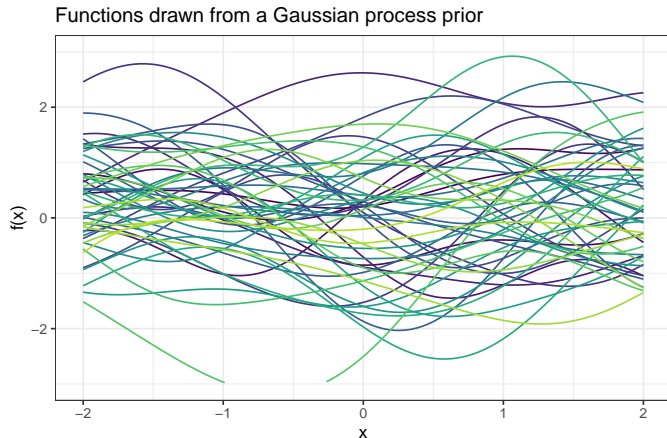
Let us change our point of view:

- Instead of “searching” for a parameter  $\theta$  in the parameter space, we directly search in a space of “allowed” functions  $\mathcal{H}$ .
- We still use Bayesian inference, but instead specifying a prior distribution over a parameter, we specify a prior distribution **over functions** and update it according to the data points we have observed.

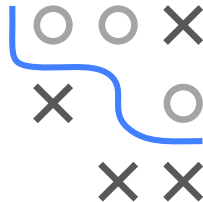


# FUNCTION-SPACE VIEW

Intuitively, imagine we could draw a huge number of functions from some prior distribution over functions (\*).

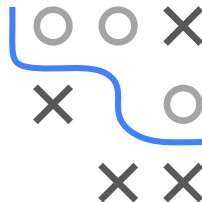
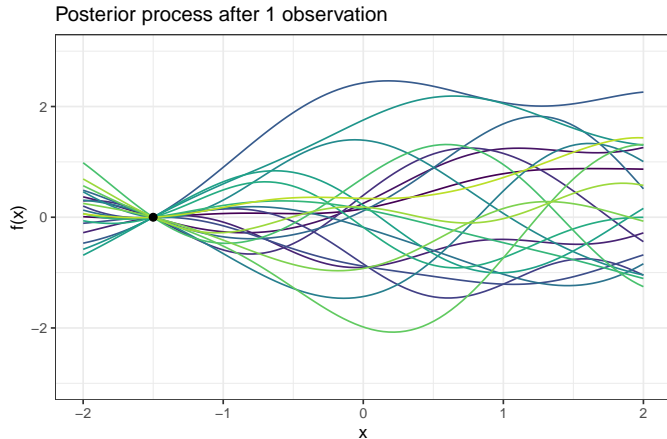


(\*) We will see in a minute how distributions over functions can be specified.



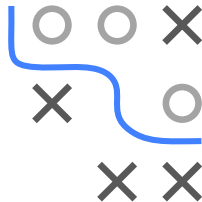
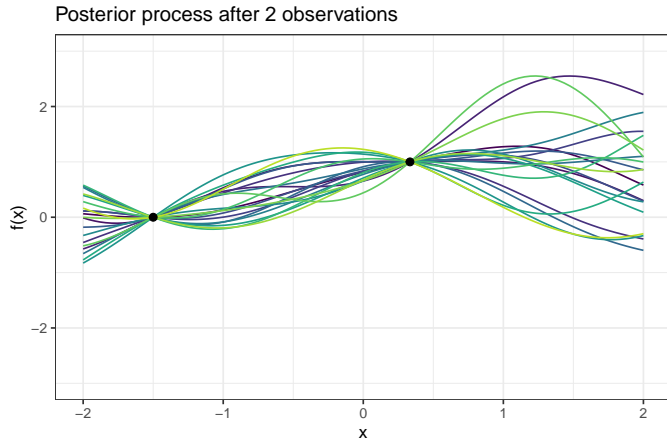
# FUNCTION-SPACE VIEW

After observing some data points, we are only allowed to sample those functions, that are consistent with the data.



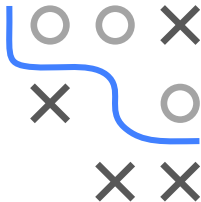
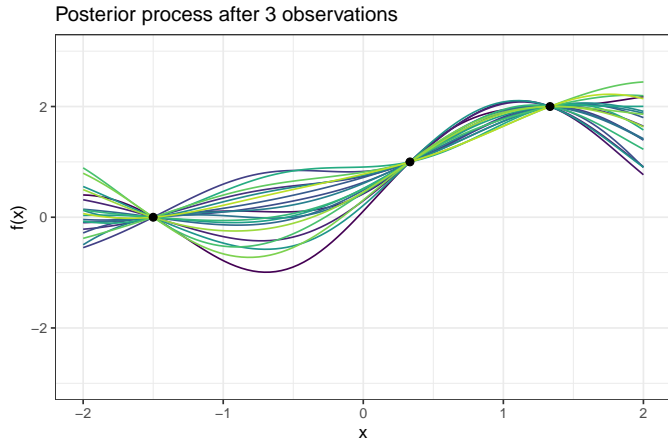
# FUNCTION-SPACE VIEW

After observing some data points, we are only allowed to sample those functions, that are consistent with the data.



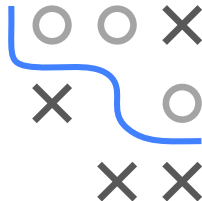
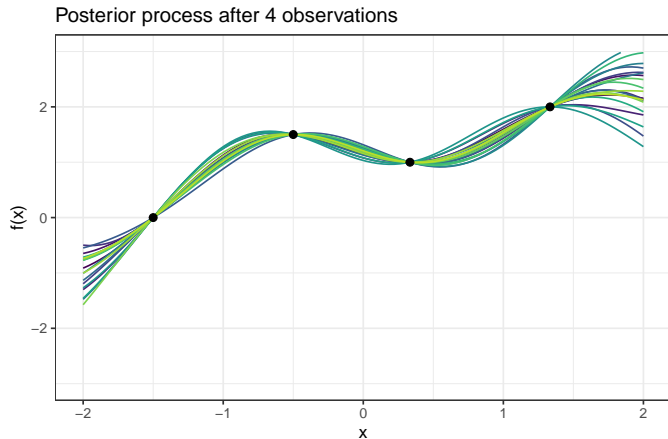
# FUNCTION-SPACE VIEW

After observing some data points, we are only allowed to sample those functions, that are consistent with the data.



# FUNCTION-SPACE VIEW

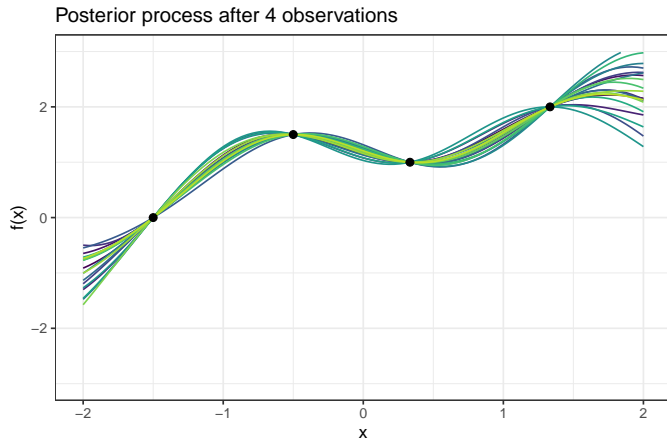
As we observe more and more data points, the variety of functions consistent with the data shrinks.





# FUNCTION-SPACE VIEW

Intuitively, there is something like “mean” and a “variance” of a distribution over functions.



# WEIGHT-SPACE VS. FUNCTION-SPACE VIEW

## Weight-Space View

Parameterize functions

Example:  $f(\mathbf{x} | \theta) = \theta^\top \mathbf{x}$

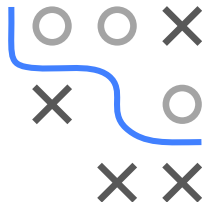
Define distributions on  $\theta$

Inference in parameter space  $\Theta$

## Function-Space View

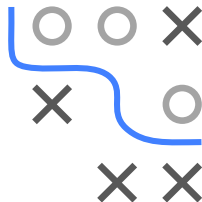
Define distributions on  $f$

Inference in function space  $\mathcal{H}$



Next, we will see how we can define distributions over functions mathematically.

# Distributions on Functions



# DISCRETE FUNCTIONS

For simplicity, let us consider functions with finite domains first.

Let  $\mathcal{X} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$  be a finite set of elements and  $\mathcal{H}$  the set of all functions from  $\mathcal{X} \rightarrow \mathbb{R}$ .

Since the domain of any  $h(\cdot) \in \mathcal{H}$  has only  $n$  elements, we can represent the function  $h(\cdot)$  compactly as a  $n$ -dimensional vector

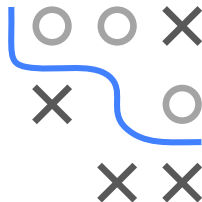
$$\mathbf{h} = \left[ h(\mathbf{x}^{(1)}), \dots, h(\mathbf{x}^{(n)}) \right].$$



# DISCRETE FUNCTIONS

**Example 1:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **two** points  $\mathcal{X} = \{0, 1\}$ .

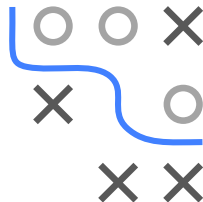
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 1:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **two** points  $\mathcal{X} = \{0, 1\}$ .

Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 1:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **two** points  $\mathcal{X} = \{0, 1\}$ .

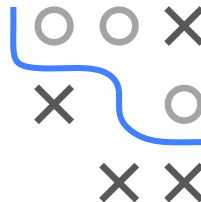
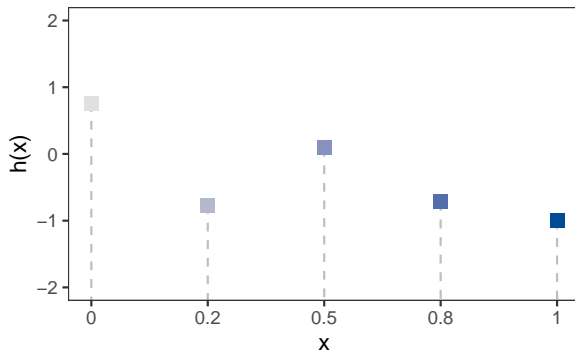
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 2:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points  $\mathcal{X} = \{0, 0.25, 0.5, 0.75, 1\}$ .

Examples for functions that live in this space:

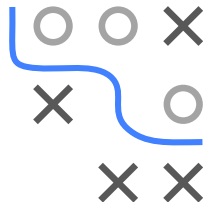
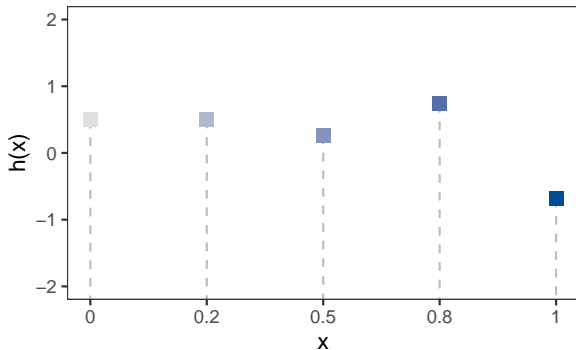




# DISCRETE FUNCTIONS

**Example 2:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points  $\mathcal{X} = \{0, 0.25, 0.5, 0.75, 1\}$ .

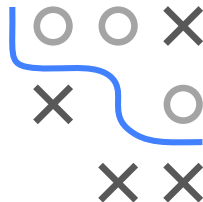
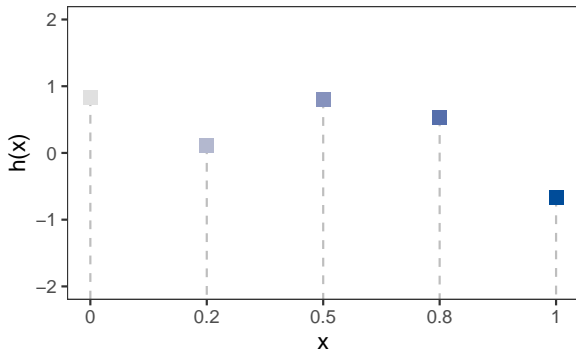
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 2:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points  $\mathcal{X} = \{0, 0.25, 0.5, 0.75, 1\}$ .

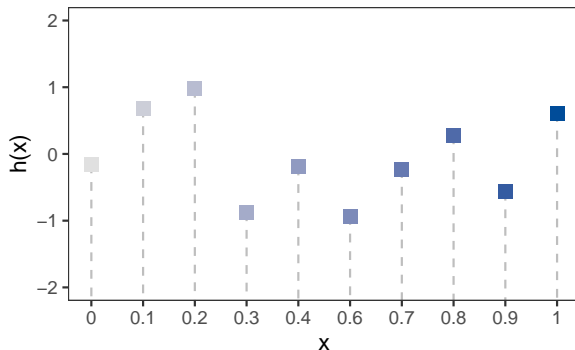
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 3:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **ten** points.

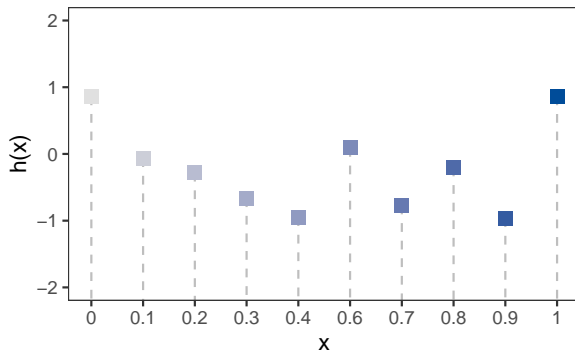
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 3:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **ten** points.

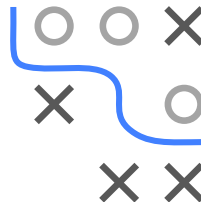
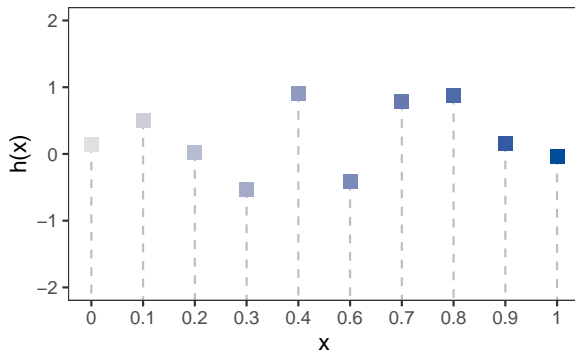
Examples for functions that live in this space:



# DISCRETE FUNCTIONS

**Example 3:** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **ten** points.

Examples for functions that live in this space:



# DISTRIBUTIONS ON DISCRETE FUNCTIONS

One natural way to specify a probability function on discrete function  $h \in \mathcal{H}$  is to use the vector representation

$$\mathbf{h} = \left[ h(\mathbf{x}^{(1)}), h(\mathbf{x}^{(2)}), \dots, h(\mathbf{x}^{(n)}) \right]$$

of the function.

Let us see  $\mathbf{h}$  as a  $n$ -dimensional random variable. We will further assume the following normal distribution:

$$\mathbf{h} \sim \mathcal{N}(\mathbf{m}, \mathbf{K}).$$

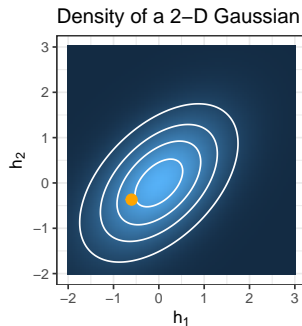
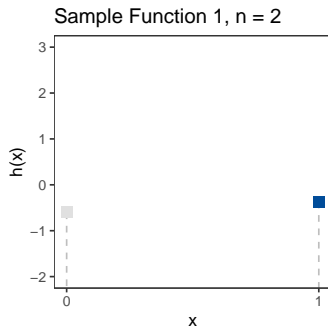
**Note:** For now, we set  $\mathbf{m} = \mathbf{0}$  and take the covariance matrix  $\mathbf{K}$  as given. We will see later how they are chosen / estimated.



# DISCRETE FUNCTIONS

**Example 1 (continued):** Let  $h : \mathcal{X} \rightarrow \mathcal{Y}$  be a function that is defined on **two** points  $\mathcal{X}$ . We sample functions by sampling from a two-dimensional normal variable

$$\mathbf{h} = [h(1), h(2)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$

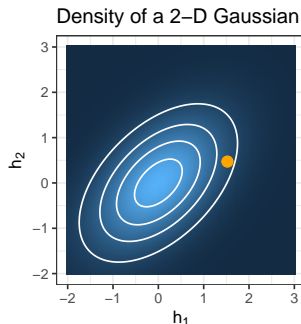
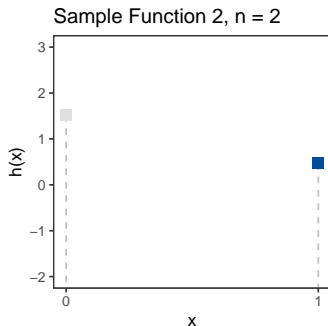


In this example,  $\mathbf{m} = (0, 0)$  and  $\mathbf{K} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$ .

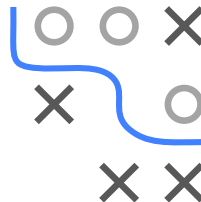
# DISCRETE FUNCTIONS

**Example 1 (continued):** Let  $h : \mathcal{X} \rightarrow \mathcal{Y}$  be a function that is defined on **two** points  $\mathcal{X}$ . We sample functions by sampling from a two-dimensional normal variable

$$\mathbf{h} = [h(1), h(2)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$



In this example,  $\mathbf{m} = (0, 0)$  and  $\mathbf{K} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$ .

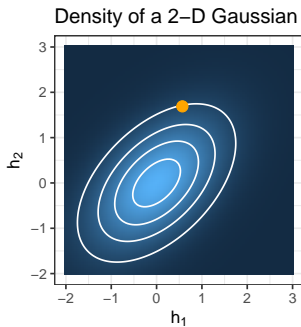
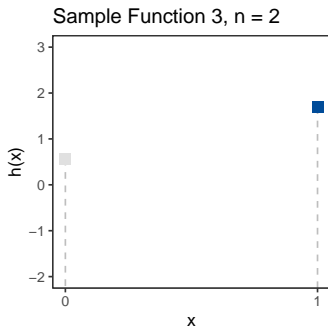




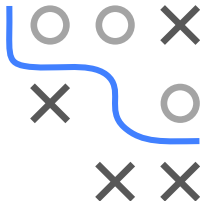
# DISCRETE FUNCTIONS

**Example 1 (continued):** Let  $h : \mathcal{X} \rightarrow \mathcal{Y}$  be a function that is defined on **two** points  $\mathcal{X}$ . We sample functions by sampling from a two-dimensional normal variable

$$\mathbf{h} = [h(1), h(2)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$



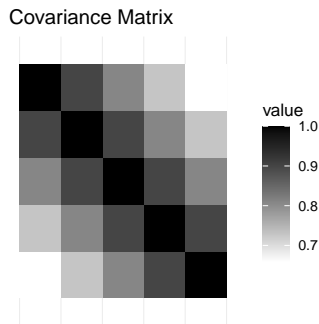
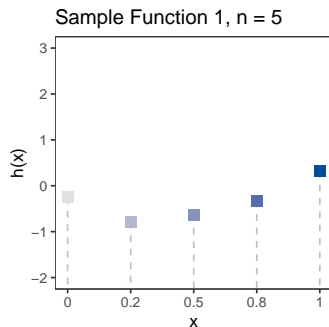
In this example,  $m = (0, 0)$  and  $K = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$ .



# DISCRETE FUNCTIONS

**Example 2 (continued):** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points. We sample functions by sampling from a five-dimensional normal variable

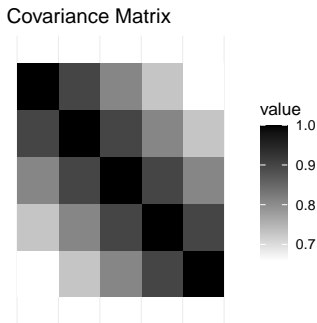
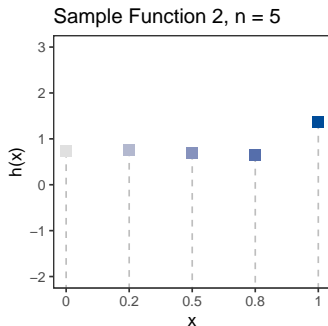
$$\mathbf{h} = [h(1), h(2), h(3), h(4), h(5)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$



# DISCRETE FUNCTIONS

**Example 2 (continued):** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points. We sample functions by sampling from a five-dimensional normal variable

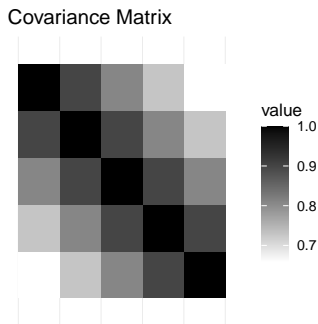
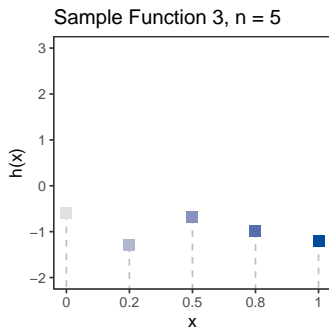
$$\mathbf{h} = [h(1), h(2), h(3), h(4), h(5)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$



# DISCRETE FUNCTIONS

**Example 2 (continued):** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **five** points. We sample functions by sampling from a five-dimensional normal variable

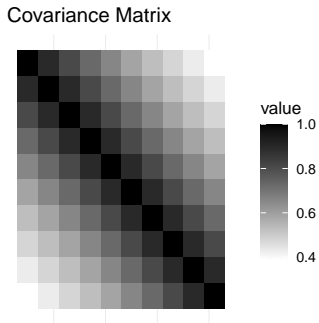
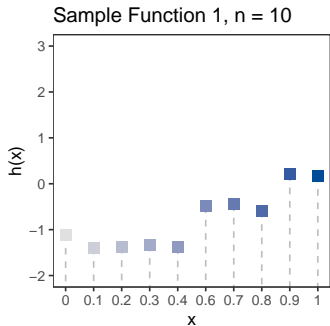
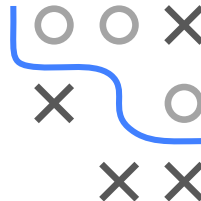
$$\mathbf{h} = [h(1), h(2), h(3), h(4), h(5)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$



# DISCRETE FUNCTIONS

**Example 3 (continued):** Let us consider  $h : \mathcal{X} \rightarrow \mathcal{Y}$  where the input space consists of **ten** points. We sample functions by sampling from ten-dimensional normal variable

$$\mathbf{h} = [h(1), h(2), \dots, h(10)] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$







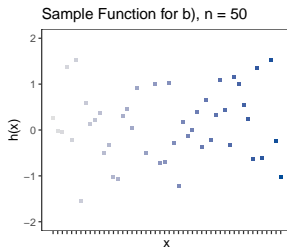
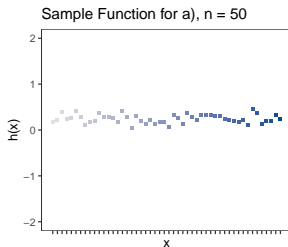
# ROLE OF THE COVARIANCE FUNCTION

Note that the covariance controls the “shape” of the drawn function.

Consider two extreme cases where function values are

a) strongly correlated:  $\mathbf{K} = \begin{pmatrix} 1 & 0.99 & \dots & 0.99 \\ 0.99 & 1 & \dots & 0.99 \\ 0.99 & 0.99 & \ddots & 0.99 \\ 0.99 & \dots & 0.99 & 1 \end{pmatrix}$

b) uncorrelated:  $\mathbf{K} = \mathbf{I}$





# ROLE OF THE COVARIANCE FUNCTION

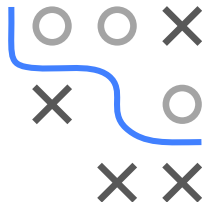
- “Meaningful” functions (on a numeric space  $\mathcal{X}$ ) may be characterized by a spatial property:

If two points  $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$  are close in  $\mathcal{X}$ -space, their function values  $f(\mathbf{x}^{(i)}), f(\mathbf{x}^{(j)})$  should be close in  $\mathcal{Y}$ -space.

In other words: If they are close in  $\mathcal{X}$ -space, their functions values should be **correlated**!

- We can enforce that by choosing a covariance function with

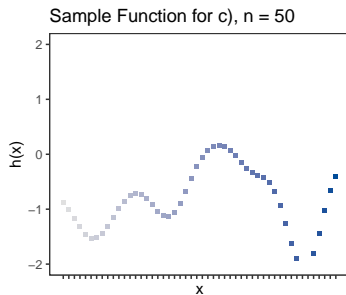
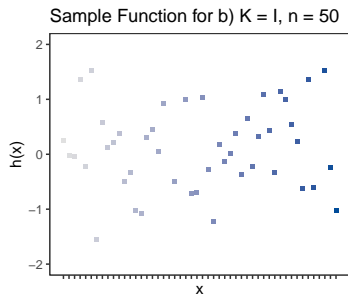
$K_{ij}$  high, if  $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$  close.



# ROLE OF THE COVARIANCE FUNCTION

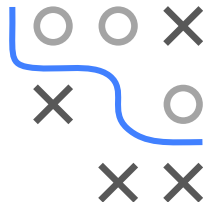
- We can compute the entries of the covariance matrix by a function that is based on the distance between  $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$ , for example:

c) Spatial correlation:  $K_{ij} = k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \exp\left(-\frac{1}{2} \|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\|^2\right)$



**Note:**  $k(\cdot, \cdot)$  is known as the **covariance function** or **kernel**. It will be studied in more detail later on.

# Gaussian Processes

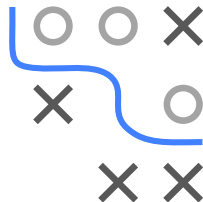
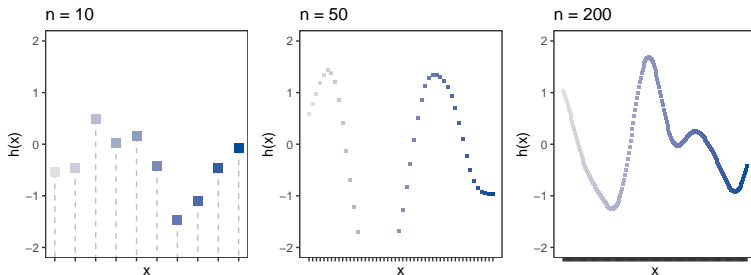


## FROM DISCRETE TO CONTINUOUS FUNCTIONS

- We defined distributions on functions with discrete domain by defining a Gaussian on the vector of the respective function values

$$\mathbf{h} = [h(\mathbf{x}^{(1)}), h(\mathbf{x}^{(2)}), \dots, h(\mathbf{x}^{(n)})] \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$

- We can do this for  $n \rightarrow \infty$  (as “granular” as we want)



# FROM DISCRETE TO CONTINUOUS FUNCTIONS

- No matter how large  $n$  is, we are still considering a function over a discrete domain.
- How can we extend our definition to functions with **continuous domain**  $\mathcal{X} \subset \mathbb{R}$ ?



# GAUSSIAN PROCESSES: INTUITION

- Intuitively, a function  $f$  drawn from **Gaussian process** can be understood as an “infinite” long Gaussian random vector.
- It is unclear how to handle an “infinite” long Gaussian random vector!



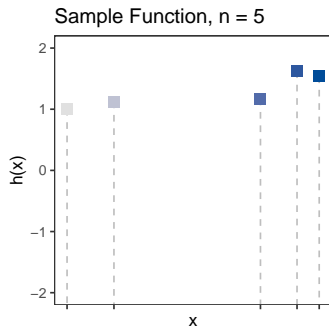
# GAUSSIAN PROCESSES: INTUITION

- Thus, it is required that for **any finite set** of inputs  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\} \subset \mathcal{X}$ , the vector  $\mathbf{f}$  has a Gaussian distribution

$$\mathbf{f} = \left[ f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)}) \right] \sim \mathcal{N}(\mathbf{m}, \mathbf{K}),$$

with  $\mathbf{m}$  and  $\mathbf{K}$  being calculated by a mean function  $m(\cdot)$  / covariance function  $k(\cdot, \cdot)$ .

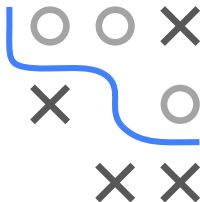
- This property is called **Marginalization Property**.



$f(x)$



$$\sim \mathcal{N}(\mu, \Sigma)$$



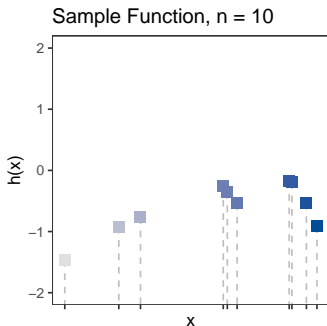
# GAUSSIAN PROCESSES: INTUITION

- Thus, it is required that for **any finite set** of inputs  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\} \subset \mathcal{X}$ , the vector  $\mathbf{f}$  has a Gaussian distribution

$$\mathbf{f} = \left[ f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)}) \right] \sim \mathcal{N}(\mathbf{m}, \mathbf{K}),$$

with  $\mathbf{m}$  and  $\mathbf{K}$  being calculated by a mean function  $m(\cdot)$  / covariance function  $k(\cdot, \cdot)$ .

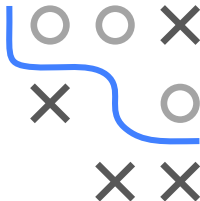
- This property is called **Marginalization Property**.



$f(x)$



$$\sim \mathcal{N}(\mu, \Sigma)$$





# GAUSSIAN PROCESSES: INTUITION

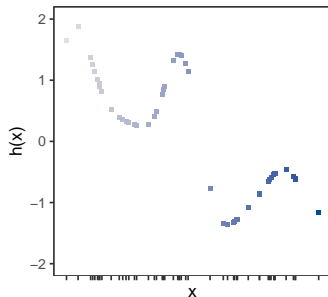
- Thus, it is required that for **any finite set** of inputs  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\} \subset \mathcal{X}$ , the vector  $\mathbf{f}$  has a Gaussian distribution

$$\mathbf{f} = \left[ f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)}) \right] \sim \mathcal{N}(\mathbf{m}, \mathbf{K}),$$

with  $\mathbf{m}$  and  $\mathbf{K}$  being calculated by a mean function  $m(\cdot)$  / covariance function  $k(\cdot, \cdot)$ .

- This property is called **Marginalization Property**.

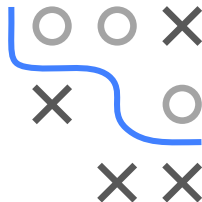
Sample Function,  $n = 50$



$f(x)$



$$\sim \mathcal{N}(\mu, \Sigma)$$



# GAUSSIAN PROCESSES

This intuitive explanation is formally defined as follows:

A function  $f(\mathbf{x})$  is generated by a GP  $\mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$  if for **any finite** set of inputs  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$ , the associated vector of function values  $\mathbf{f} = (f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)}))$  has a Gaussian distribution

$$\mathbf{f} = \left[ f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(n)}) \right] \sim \mathcal{N}(\mathbf{m}, \mathbf{K}),$$

with

$$\mathbf{m} := \left( m(\mathbf{x}^{(i)}) \right)_i, \quad \mathbf{K} := \left( k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \right)_{i,j},$$

where  $m(\mathbf{x})$  is called mean function and  $k(\mathbf{x}, \mathbf{x}')$  is called covariance function.



# GAUSSIAN PROCESSES

A GP is thus **completely specified** by its mean and covariance function

$$\begin{aligned}m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}\left[(f(\mathbf{x}) - \mathbb{E}[f(\mathbf{x})]) (f(\mathbf{x}') - \mathbb{E}[f(\mathbf{x}')])\right]\end{aligned}$$

**Note:** For now, we assume  $m(\mathbf{x}) \equiv 0$ . This is not necessarily a drastic limitation - thus it is common to consider GPs with a zero mean function.



# SAMPLING FROM A GAUSSIAN PROCESS PRIOR

We can draw functions from a Gaussian process prior. Let us consider  $f(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$  with the squared exponential covariance function <sup>(\*)</sup>

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{1}{2\ell^2}\|\mathbf{x} - \mathbf{x}'\|^2\right), \quad \ell = 1.$$

This specifies the Gaussian process completely.



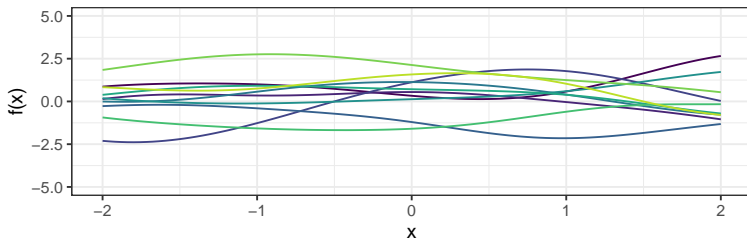
<sup>(\*)</sup> We will talk later about different choices of covariance functions.

# SAMPLING FROM A GAUSSIAN PROCESS PRIOR

To visualize a sample function, we

- choose a high number  $n$  (equidistant) points  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$
- compute the corresponding covariance matrix  $\mathbf{K} = (k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}))_{i,j}$  by plugging in all pairs  $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$
- sample from a Gaussian  $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$ .

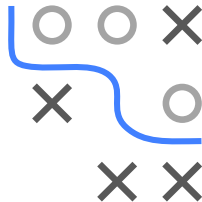
We draw 10 times from the Gaussian, to get 10 different samples.



Since we specified the mean function to be zero  $m(\mathbf{x}) \equiv 0$ , the drawn functions have zero mean.



# Gaussian Processes as Indexed Family



# GAUSSIAN PROCESSES AS AN INDEXED FAMILY

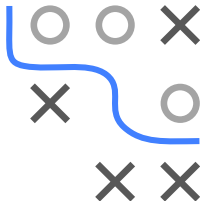
A Gaussian process is a special case of a **stochastic process** which is defined as a collection of random variables indexed by some index set (also called an **indexed family**). What does it mean?

An **indexed family** is a mathematical function (or “rule”) to map indices  $t \in T$  to objects in  $\mathcal{S}$ .

## Definition

A **family of elements in  $\mathcal{S}$  indexed by  $T$**  (indexed family) is a surjective function

$$\begin{aligned}s : T &\rightarrow \mathcal{S} \\ t &\mapsto s_t = s(t)\end{aligned}$$







# INDEXED FAMILY

But the indexed set  $\mathcal{S}$  can be something more complicated, for example functions or **random variables** (RV):

- $T = \{1, \dots, m\}$ ,  $Y_t$ 's are RVs: Indexed family is a random vector.
- $T = \{1, \dots, m\}$ ,  $Y_t$ 's are RVs: Indexed family is a stochastic process in discrete time
- $T = \mathbb{Z}^2$ ,  $Y_t$ 's are RVs: Indexed family is a 2D-random walk.

